10/542210

 JC20 Rec'd PCT/PTO 14 JUL 2005

Description

Method and arrangement for suppressing incorrect messages in monitoring systems

The invention relates to a method and an arrangement for suppressing incorrect messages in monitoring systems for electronic devices, in particular for sensor circuits for motor vehicles.

For safety reasons, monitoring systems are frequently used for electronic devices in which an incorrect output variable which is caused by a fault in the device can bring about hazards. It is therefore necessary to avoid faults in sensor circuits for motor vehicles resulting in hazardous driving situations, for example if a rotational speed sensor signals a high rotational rate while the vehicle is maintaining its course. However, other hazards and at least operational faults may occur as a result of incorrect messages. Causes of incorrect messages may be, for example, brief disruptions, in particular voltage peaks, which are interpreted by monitoring systems as faults without them leading to falsification of the output variable of the device.

The object of the present invention is to largely prevent incorrect messages so that as far as possible only genuine faults lead to an alarm, which is then indicated or can be fed to a superordinate system in order to ignore the output variable.

This object is achieved according to the invention in that fault messages increment a counter, and in that an alarm is not triggered until a predefined counter reading is

reached. There is preferably provision here for the counter to be decremented according to time periods without a fault message.

By means of one development of the method in which there is provision for the size of the increments and, if appropriate, of the decrements and the predefined counter reading to be preselectable, the triggering of an alarm can be adapted individually to the type of respective fault message. This development is preferably implemented by the preselectable variables being read out from a nonvolatile memory when the device is switched on.

In devices which are to be monitored it is generally expedient to monitor a plurality of variables, referred to below as input variables with respect to the method according to the invention. For this purpose, in the method according to the invention there is provision for fault messages which each increment a counter to be derived from a plurality of input variables to be monitored, and for the size of the increments and, if appropriate, of the decrements, the predefined counter reading and limiting values of the variable to be respectively monitored to be preselectable for each of the input variables to be monitored.

In the case of devices which are relevant to safety it is frequently necessary to react very quickly to a fault message. The checking of the fault messages with the method according to the invention can, however, take longer than necessary, in particular when there are a plurality of input variables to be monitored, due to the finite running time of the program in the processor. This applies in particular if a plurality of incrementations are necessary up to the point where the predefined counter reading is reached, in order to avoid incorrect messages.

In one development of the method according to the invention the triggering of an alarm is speeded up significantly in that fault messages are derived using a program running on a processor if one of the input variables to be monitored exceeds respective limiting values which are predefined for it, in that the respective counter is incremented by the fault messages, in that the counter reading is checked to determine whether at least one fault message is present, and in that, if this is the case, a further check is carried out in advance for the relevant input variable. This is achieved in that the monitoring of a variable which has already had attention drawn to it by a fault message is carried out in a prioritized fashion in terms of time. If a plurality of such fault messages are present, the further checking will preferably be carried out in such a way that when fault messages are present for a plurality of input variables the advance further checking of these input variables is carried out according to a previously defined priority list.

The invention also comprises an arrangement for suppressing fault messages in monitoring systems for electronic devices, in particular for sensor circuits for motor vehicles, in which it is provided that in a microprocessor a program it is possible to run with which fault messages are derived if one of the input variables to be monitored exceeds respective limiting values which are predefined for it, and with which in addition the respective counter is incremented by the fault messages, the counter reading is checked to determine whether at least one fault message is present, and, if this is the case, a further check is carried out in advance for the relevant input variable.

The invention permits numerous embodiments. One of these is illustrated schematically in the drawing by means of a plurality of figures, and described below. In the drawing:

figure 1 is a block diagram of the exemplary embodiment,

figure 2 comprises time diagrams for various signals in the block diagram illustrated in fig. 1, and

figure 3 is a flowchart of a program which executes the method according to the invention.

For the sake of simplicity, the arrangement according to figure 1 is illustrated only for an input variable which is to be monitored and which is present at an input 1. At first, checking is carried out at 2 to determine whether the input variable is in a permissible range between min and max. If this is not the case, a fault message is passed on to a fault counter 3, also simply referred to below as counter. If a fault message is present, the counter 3 is incremented by one increment I which is predefined as a number of counting steps. If no fault message is present in the respective time period (program run), the counter 3 is decremented by one decrement D which is also predefined as a number of counter steps. If the counter reading reaches a predefined threshold value A, an alarm is triggered in an alarm trigger 4 and output via an output 5. The respective counter reading can be read out via a monitoring output 6.

When the device is switched on, the predefined values are read out from an EEPROM 7, said values being specifically the maximum permissible value max and the minimum permissible value min of the input signal, the number of counting steps I and D and the threshold value A. In addition,

the counter 3 is reset by means of an input 8. The part of figure 1 which is outlined by dashed lines is present once for each input variable and is preferably implemented by a program for a processor, with the predefined variables being stored individually for each input variable in the EEPROM 7 and being loaded into the working memory of the processor when the device is switched on. The outputs 5, 6 which are assigned to the alarm triggers for the various input variables can be combined in a suitable way.

By means of the threshold values it is possible to suppress the alarm in a selective way when specific input variables to be monitored are exceeded in that the threshold value is set to "0" for these input variables. This specification of a threshold value is interpreted as a command with which the correspondingly configured alarm triggers are blocked. The fault counter for checking these input variables thus has no effect any more on the triggering of the alarm.

Figure 2 shows an exemplary profile of the output variable of the range checking 2 in line a. As an example it will be assumed that after a fault-free time a fault message 11 occurs and then in turn a time period without faults and then two fault messages 12, 13 in succession.

Line b shows the profile of the counter reading for the case in which in each case a fault message 11, 12, 13 increments the counter by a counting step, and in which the decrementation is set to D=0 in the case of the absence of a fault message. With this setting, no fault messages are therefore "forgotten" again so that the absolute number of fault messages remains stored until the next reset (switching on of the device or switching off). If the threshold value is therefore

set to 3 in the case illustrated in figure 2b, an alarm is triggered on the basis of the fault message 13.

Figure 2c relates in turn to the occurrence of fault messages according to figure 2a, but I is set to 3 and D to 1. As a result of fault messages occurring in close succession, the counter reading quickly rises, and when fault messages are absent it drops more slowly. As a result, accumulations of fault messages lead to an alarm if the threshold value A is set higher than D.

The program illustrated in part in figure 3 runs through a loop during which all the input variables from 1 to n are checked. In the program steps 21, 22, the first two input variables are checked, the respective counter is incremented or decremented and the counter reading is compared with the associated threshold value A. There is then an interrogation to determine whether one of the input variables has incremented the counter 3 as a result of the limiting values max and min (figure 1) being respectively exceeded or undershot, which is considered to be a warning. If this is not the case, after the branching point 23 the program continues with checking further input variables. However, if at least one warning is present, the relevant input variables in the program part 24 are checked again. This takes place according to a previously defined priority sequence.

If it is detected during one of the checks 21, 22, 24 that the alarm threshold A has been reached, an alarm is triggered, which is not illustrated in particular in figure 3 for the sake of clarity. The last two input variables (n-1) and n are checked at 25 and 26, after which in turn branching takes place as a function of whether these

input variables have lead to a warning. If this is the case, the relevant input variables are checked again at 28. The program is then repeated starting at 21.

The described checking of two input variables including the immediate further checking at 24 takes approximately 5 ms when a program is carried out in practice. If a maximum reaction time for a fault message of 25 ms is allowed, ten input variables can be checked. The order of the checks can also be changed depending on the particular requirements. For example, three or more input variables can thus be checked before the system continues to immediate further checking of those input variables for which a warning is active. In other cases it is also possible to provide that after each input variable has been checked it is decided whether said input variable is to be checked once more immediately afterward.